WEST Search History

Hide Items | Restore | Clear | Cancel

DATE: Monday, July 19, 2004

Hide?	Set Name	Query	Hit Count
		DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=ADJ	
☐	L5	20001221	19
☐	L4	L3 and ((storage or memory) near8 management)	100
☐	L3	L2 and (launcher or controller)	577
☐	L2	((mobile near8 (device or telephone)) or (PDA or (personal adj2 digital))) same ((limited or limitted or limit) near8 (storage or capacity or memory or cache))	1666
☐	L1	((mbile near8 (device or telephone)) or (PDA or (personal adj2 digital))) same ((limited or limitted or limit) near8 (storage or capacity or memory or cache))	860

END OF SEARCH HISTORY

WEST Search History

Hide Items | Restore | Clear | Cancel

DATE: Monday, July 19, 2004

Hide?	Set Name	Query	Hit Count
		DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=ADJ	
☐	L15	((local near5 memory)near8 (second near5 (memory or cache or storage))) same (mobile near8 (device or telephone or PDA or (personal adj2 digital)))	0
☐	L14	(program near5 size) near8 (priority) near8 (time near5 stored) near8 (use near4 time)	0
☐	L13	20001221	11
☐	L12	L11 and l9	0
☐	L11	(control near5 parameter) near8 (storage near5 configuration)	24
☐	L10	L9 and (external adj4 (storage or memory))	17
☐	L9	20001221	641
☐	L8	((mobile near8 (device or telephone)) or (PDA or (personal adj2 digital))) near8 ((storage or capacity or memory or cache) near8 (management or control))	1357
☐	L7	20001221	1365
☐	L6	((mobile near8 (device or telephone)) or (PDA or (personal adj2 digital))) same ((storage or capacity or memory or cache) near8 (management or control))	3165
☐	L5	20001221	19
☐	L4	L3 and ((storage or memory) near8 management)	100
☐	L3	L2 and (launcher or controller)	577
☐	L2	((mobile near8 (device or telephone)) or (PDA or (personal adj2 digital))) same ((limited or limitted or limit) near8 (storage or capacity or memory or cache))	1666
☐	L1	((mbile near8 (device or telephone)) or (PDA or (personal adj2 digital))) same ((limited or limitted or limit) near8 (storage or capacity or memory or cache))	860

END OF SEARCH HISTORY

Previous Doc Next Doc Go to Doc#
First Hit Fwd Refs

Search Forms
Search Results
Help
User Searches
Preferences
Logout

Generate Collection

L5: Entry 6 of 19 File: USPT Oct 29, 2002

DOCUMENT-IDENTIFIER: US 6473609 B1
TITLE: Method and architecture for interactive two-way communication devices to interact with a network

Abstract Text (1):
The present invention is particularly applicable to navigation of Internet by two-way interactive communication mobile devices that are capable of wireless communication via a link server with service providers or network servers on the Internet. Despite the limited computing resources in mobile devices that make it economically and technically impractical for the mobile devices to operate a local browser functioning as if it was in a desktop computer, the present invention allows the mobile devices to interact effectively with the Internet using a control engine operating in the link server and an interface engine operating in the mobile devices. The control engine, which utilizes the computing resources of the link server device, is responsible for tasks that require considerable computing power and memory, such as processing of URL requests, interpretation of markup language files, management of data cache and variable states. Further, working with a message processor in the server device, the control engine communicates with an interface engine using a compact data format that is efficiently transportable in the wireless data network. The interface engine typically performs tasks that do not require considerable computing power and memory, such as receiving input data from users, and the rendering of the compact data format received from the link server device, to cause the mobile device to display contents in the markup language files on a display screen.

Application Filing Date (1):
19980914

Brief Summary Text (12):
The present invention addresses the above described problems and is particularly applicable to navigation of Internet web pages by two-way interactive communication mobile devices (e.g., mobile computing devices, cellular phones, palm-sized computer devices, personal digital assistant devices and Internet-capable appliance remote controllers) which are capable of wireless communication via a link server with service providers or network servers on the Internet. Despite the common deficiencies of mobile devices (i.e., a primitive processor, little memory and limited graphics capability) which make it economically and technically impractical for the mobile devices to operate a local browser functioning as if it was in a desktop computer, the present invention allows the mobile devices to interact effectively with the Internet and can be used with a wide variety of wireless communication networks (e.g., cellular digital packet data (CDPD) network, Global System for Mobile Communications (GSM) network, Code Division Multiple Access (CDMA) network and Time Division Multiple Access (TDMA) network).

Brief Summary Text (13):
According to one aspect of the present invention, a mobile device includes an interface engine that, via a client module, communicates and operates with a control engine in a link server device over a wireless network. The control engine, which utilizes the computing resources of the link server device, is responsible

for tasks that require considerable computing power and memory, such as processing of URL requests, interpretation of markup language files, management of data cache and variable states. Further, working with a message processor in the server device, the control engine communicates with an interface engine using a compact data format that is efficiently transportable in the wireless data network. The interface engine typically performs tasks that do not require considerable computing power and memory, such as receiving input data from users, and the rendering of the compact data format received from the link server device, to cause the mobile device to display contents in the markup language files on a display screen.

Detailed Description Text (3):
There are n mobile devices 106 serviced by airnet 102. Mobile devices 106 are interactive two-way communication devices (e.g., mobile computing devices, cellular phones, palm-sized computing devices with PDA (Personal Data Assistants) functionality and Internet-capable appliance remote controllers) which are capable of communicating wirelessly with antenna 108 via airnet 102. As shown, antenna 108 also represents a wireless carrier infrastructure that generally includes a base station and an operations and maintenance center. The base station controls radio or telecommunication links with mobile devices 106. The operations and maintenance center comprises a mobile switching center performing the switching of calls between the mobile devices and other fixed or mobile network users. Further the operations and maintenance center manages mobile account services, such as authentication, and oversees the proper operation and setup of the wireless network. Each of the hardware components and processes in carrier infrastructure 108 are known to those skilled in the art and thus are not described here to avoid unnecessarily obscuring aspects of the present invention.

Detailed Description Text (8):
Nevertheless, cellular phone 200 or mobile devices 106 of FIG. 1 typically do not have the computing resources to implement HTTP to run an HTML browser. The computing power in cellular phone 200 or mobile devices 106 of FIG. 1 is typically less than one percent of a laptop personal computer's computing power, the memory capacity is generally less than 128 kilobytes and the graphics display capability is very limited. Cellular phone 200 or any of mobile devices 106 of FIG. 1 is not a replacement of a desktop computing device or the combination of a wireless communication module and a personal computer. Further, making a mobile device, such as cellular phone 200, capable of navigating hypermedia information in a network server is a significant departure from prior art systems.

Detailed Description Text (12):
HDTP is a session-level protocol that resembles HTTP but runs on UDP and without incurring the overhead of HTTP/TCP and is highly optimized for use in thin devices, such as the mobile devices, that have significantly less computing power and memory than those of a desktop personal computer. Further, UDP does not require a connection to be established between a client device and a server before information can be exchanged, which eliminates the need of exchanging a large number of packets during a session creation. Exchanging a very small number of packets during a transaction is one of the desired features for a mobile device with limited computing power and memory to effectively interact with a landline device.

Detailed Description Text (56):
As described above, screen display 716 also includes the representations of two soft keys, an OK key 706, and a Back key 714. In this example, these soft keys are defined only for the card used to generate screen display 716. The "OK" key allows the user to proceed with the chosen item and the "Back" soft key allows the user to go back the previous screen display if so desired. In the present invention, the "Back" soft key may generate a request that is sent over to the link server from which the previous screen display is fetched again. Other keys can be implemented.

Record Display Form

For example a "Home" key, resulting in a request that returns the user to screen display 708 of FIG. 7B. The "Home" key may be associated with a resource locator identifying the card representing screen display 708. Specifically, the link server manages a limited history stack of recent requests made by the mobile device in a memory. When a request is made, the control engine looks up the history stack to see if the request is an "old" one. For example, when the "Home" key is pressed, the request can be found in the history stack and the contents, either in the form of an HDML card or an SDD file, can be retrieved from memory and forwarded to the mobile device for display.

Previous Doc Next Doc Go to Doc#

Previous Doc Next Doc Go to Doc#
First Hit Fwd Refs

Generate Collection

L5: Entry 13 of 19 File: USPT Mar 21, 2000

DOCUMENT-IDENTIFIER: US 6041374 A
TITLE: PCMCIA interface card for coupling input devices such as barcode scanning engines to personal digital assistants and palmtop computers

Application Filing Date (1):
19970314

Brief Summary Text (12):
In one embodiment, the signal from an undecoded barcode scan engine or other input device that outputs electrical signals that encode alphanumeric characters is coupled to a specially programmed decoder chip on the PC card. The barcode scan engine or other input device is either external to the PC card or physically mounted thereon. The decoder chip decodes the electrical signals into alphanumeric characters and generates an interrupt to the host computer through a pin on the PCMCIA bus. The host computer then does an I/O transaction to the PC card to retrieve the decoded data. In one particularly useful species of this genus, the PC card includes nonvolatile memory which may be accessed by the host computer through the PCMCIA bus without blocking access to the decoder through the PCMCIA bus. In this way, host computers that are memory limited like PDA's may replace their PC memory card with a barcode decoder PC card having on-board nonvolatile memory and have the benefit of both PC card barcode decoding (or access to data from other types of input devices) while not losing the benefit of also have external nonvolatile memory which may be used for any purpose.

Drawing Description Text (34):
FIG. 33 is a block diagram of the flow of the program that programs the GAL logic configuration for a gate array logic controller in the hardware embodiment shown in FIG. 36.

Detailed Description Text (64):
Note that RAM 50 can have more capacity than is needed simply to implement the PCMCIA interface. This allows the bar code scanning engine interface to have the additional function as serving as a flash memory card for the PDA since many PDA and palmtop devices are severely limited in memory capacity and need more to run complex programs. Up to four megabytes of RAM can be addressed in the Common Memory Space of a PCMCIA defined PC card, but usually only two megabytes or less are required for the bar code scanning engine interface. This enables PDA and palmtop devices with only one PCMCIA slot to have the functionality of an expansion memory card in addition to a laser based bar code scanning or other input device without having to switch PC cards. Expansion memory cards of DRAM, EEPROM and EPROM types are commercially available and manufacturers thereof are listed in the PCMCIA Resource Reference Book of Spring 1994. The details of these commercially available memory expansion cards is hereby incorporated by reference.

Detailed Description Text (81):
Referring to FIG. 10, there is show a flow chart for a typical process flow to implement a PCMCIA based PC card interface for a conventional wand type barcode reader housed within housing 28 shown in FIG. 9 as attached to the PC card interface circuit 32. The circuitry of either FIG. 3 or FIG. 8, or equivalents,

including interfaces based upon the Dr. Neuhaus PCMCIA interface Controller Chip, which is commercially available from Neuhouse GMBH, the details of which are hereby incorporated by reference, may be used to implement the interface on PC card 32 or any of the other PC card interfaces disclosed herein. The software of FIG. 10 may be executed on any of these equivalent circuits.

Detailed Description Text (115):
Another advantage of a PC card interface for an input device is the ability to simultaneously expand the memory capacity of the PDA in single PC card slot PDA's. There are many PC cards currently available which serve the sole purpose of expanding the rather limited memory capacity of some PDA's or providing network connectivity or modem capability. This allows more complex processing to be performed with larger programs.

Detailed Description Text (140):
In the embodiment shown in FIG. 26 using the program of Appendix A, the nonvolatile memory 570 is available for use by the host 500 as external memory after the host is rebooted. In such a case, no decoded barcode data can be input to the host. The current state of PCMCIA socket implementations on host computers do not reliably support multifunction PC cards. The only type PC cards that are broadly supported by a wide range of host computers are certain type of flash memory cards, and modem cards or LAN interface cards. Most non-PDA hosts with PCMCIA slots do fully implement the various software and hardware layers discussed in FIG. 5, however many PDAs do not fully implement all the software layers discussed in FIG. 5. This can cause problems with operability of multifunction PC cards or PC cards other than flash memory cards, modem cards or LAN interface cards. The problem with multifunction cards that include flash memory along with other functionality such as the barcode decoding circuitry of the embodiment of FIG. 26 is that when the flash memory driver of the host 500 takes control of the PCMCIA socket for access to the memory, that precludes other driver software on the host 500 from using the PCMCIA socket to access the other functionality on the PC card. Most hosts with PCMCIA sockets come equipped with drivers that execute at boot time and look for PC cards present in the PCMCIA slot, and if they find one, they try to determine what type of card it is and then take over the PCMCIA socket and carry out operations with the PC card. These drivers supplied by the host manufacturer will find flash memory on the PC card 580 in FIG. 26 and will block all access through the PCMCIA socket to the decoded data from the barcode decoding chip. Therefore, it is necessary in the embodiment symbolized by FIG. 26 to disable these drivers and use a custom written driver for controlling PCMCIA socket 528 which will accept the decoded alphanumeric data from the decoder chip 538. In one embodiment where a DOS based host was used, the standard drivers that came with the host were disabled by commenting them out in the config.sys file. In other types of hosts, similar steps should be taken to prevent blocking of the PCMCIA slot by the standard PCMCIA drivers. The custom driver that is used in these embodiments is attached hereto as Appendix A. This driver will still not provide access by the PDA to the flash memory 570 in addition to access to the decoded alphanumeric data unless the host is rebooted, but in other embodiments a different custom driver 519 will provide access to both the nonvolatile memory 570 as well as the decoded alphanumeric data from the decoder chip 538 without rebooting. In some embodiments, this difficulty can be eliminated by eliminating the flash memory, but in the preferred embodiment the flash memory is present because most PDA's and palmtops are severely limited in onboard memory which limits their usefulness.

Detailed Description Text (155):
Therefore, to get the barcode card's interrupt number and the base address of its registers, the client has to obtain this information indirectly by interrogation of the CS layer. This done by using software interrupts. In DOS machines, there are hardware interrupts where a circuit activates an interrupt line and software interrupts where a client application executes an interrupt instruction. In the case of a hardware interrupt, an interrupt controller chip in the host places the

interrupt number on the host bus in response to an interrupt acknowledge cycle on the host bus after the interrupt line is activated. The operating system reads this interrupt number and uses it as an index into an interrupt vector table which contains the address of an appropriate interrupt service routine for each specific interrupt. In the case of a software interrupt, the interrupt instruction contains the interrupt number index into the interrupt vector table. Software interrupts are how client applications communicate with the operating system in DOS machines. The CS layer is like an extension of the operating system and is assigned to interrupt number 26.

Detailed Description Text (164):
Blocks 658 and 660 represent the process carried out by the interrupt service routine for the barcode card in retrieving decoded data after an interrupt is received from the barcode card. When the PCMCIA adapter chip 536 activates the IRQ interrupt request line, the interrupt controller chip in the host places the interrupt number assigned to the barcode card on the host bus in response to the host bus interrupt acknowledge cycle. The operating system then uses this number as an index for entry into the interrupt vector table to retrieve the start address of the appropriate interrupt service routine for the barcode card in the barcode card driver. Processing is then vectored to this service routine which executes and conducts an I/O cycle with the barcode card to retrieve whatever decoded data is stored in the barcode card's registers.

Detailed Description Text (175):
Referring to FIG. 33, there is shown a block diagram of the flow of the program that programs the GAL logic configuration for a gate array logic controller in the hardware embodiment described below. A program similar to the flow chart shown on FIG. 33 is included herewith as Appendix C. The GAL or gate array logic serves to provide the Boolean logical relationships between the various signals generated which are input to and output from the decoder chip and the PCMCIA adapter chip. Block 708 represents the process of defining which pins of the GAL chip are assigned as the various functions (signals) that are to be logically related to each other by the GAL chip. Block 710 represents the process of programming the GAL with the Boolean logic relationships that define the interrelationships between the functions assigned to the various pins of the GAL in block 708.

Detailed Description Text (185):
Referring to FIG. 37, there is shown a block diagram of a system including a host computer which does decoding on-board the host and a PC card which only passes digitized samples of the barcode pattern to the host for decoding. FIG. 37 is intended to be an adjunct to the software architecture diagram of FIG. 5 showing a typical software architecture for an embodiment where the decoding is done on the host and not on the PC card to show the relationships between the various routines controlling the host microprocessor 516 although not all the details such as the keyboard, display, input device etc are shown. The host computer 500 in this embodiment only receives sample data from the PC card and not complete alphanumeric characters. The sample data comes in through the PCMCIA socket 502 (circuits or software routines with the same reference numbers as circuits or software routines in previous drawings have the same structure and function generally) and is passed via bus 774 to a PCMCIA bus controller circuit 115 the functions to drive data onto and receive data from the PCMCIA bus 48. The barcode image data samples generated by the PC card 776 are read by the host from a register or memory in the I/O space (or memory space in some embodiments) of the PC card and are transferred over the PCMCIA bus 48 through the PCMCIA bus controller 115 into a block of memory 778 in the host RAM 95 reserved to store the sample data. The sample data may be stored in RAM 95 directly by the PCMCIA bus controller circuit 115 via DMA transactions, or it may be read by the microprocessor 516 and stored in RAM 95.

Detailed Description Text (187):
The PC card will generate an interrupt when it has stored therein a predetermined

number of samples that need to be stored in the host RAM. The processing of this interrupt request is as described above with reference to block 658 in FIG. 29B. I/O transactions between the host and PC card are carried out by the barcode client 786 using the driver 781, Card Services 784, the memory technology driver layer 792, the socket services layer 794 and the PCMCIA bus controller 115. Card services talks to the socket services layer via data path 796. Socket services 794 controls the PCMCIA bus controller circuit 115 as symbolized by dashed line 795. All data paths between various software layers or between software layers and hardware circuits are only symbolic and interprocess communication may by any known method such as interrupt, interprocess pipeline, shared memory, control and data signals etc. will suffice. The purpose of socket services is to mask the details of the particular PCMCIA bus controller 115 used so as to present a uniform programmatic interface to the CS layer 784.

Detailed Description Paragraph Table (9):

CONTROL LOGIC FOR GAL CONTROLLING BARCODE DECODER CHIP " HBCR.ABL -- Control logic for TPS HBCR-2210-based bar code decoder card. Z Initial version, Jude Miller, 2/16/95 " MODULE hbcr FLAG `-r3` TITLE `HBCR -- Controller for HP Bar Code Decoder IC` U99Z DEVICE`P20V8R`; " special purpose pins MEMW.sub.-- DUP.sub.-- PIN 1; "wired externally to MEMW.sub.-- OE.sub.-- PIN 13; " Input pins A21 PIN 2; A4 PIN 3; A3 PIN 4; MEMW.sub.-- PIN 5; MEMR.sub.-- PIN 6; HBCR.sub.-- PGB PIN 7; HBCR.sub.-- CRD.sub.-- PIN 8; HBCR.sub.-- DWR.sub.-- PIN 9; HBCR.sub.-- RTS.sub.-- PIN 10; UART.sub.-- RDA PIN 11; HBCR.sub.-- PDR PIN 14; UNUSED.sub.-- 1 PIN 23; " output pins HBCR.sub.-- WE.sub.-- PIN 22; "no feedback in complex mode HBCR.sub.-- OE.sub.-- PIN 21; D1 PIN 20; D0 PIN 19; "no input in simple mode HBCR.sub.-- DDY.sub.-- PIN 18; "no input in simple mode HBCR.sub.-- CDY.sub.-- PIN 17; IRQ PIN 16; IRQMASK PIN 15; "no feedback in complex mode " definitions, including active-high aliases for active-low pins. Note that " all active.sub.-- low pin names end in `.sub.-- `. H,L,X,Z = 1,0,.X.,.Z.; DataBus = [D1, D0]; MEMW = IMEMW.sub.-- ; MEMR = IMEMR.sub.-- ; HBCR.sub.-- DWR = IHBCR.sub.-- DWR.sub.-- ; HBCR.sub.-- CRD = IHBCR.sub.-- CRD.sub.-- ; HBCR.sub.-- RTS = IHBCR.sub.-- RTS.sub.-- ; HBCR.sub.-- WE = IHBCR.sub.-- WE.sub.-- ; HBCR.sub.-- OE = IHBCR.sub.-- OE.sub.-- ; HBCR.sub.-- DDY = IHBCR.sub.-- DDY.sub.-- ; HBCR.sub.-- CDY = IHBCR.sub.-- CDY.sub.-- ; Address = [A21, A4, A3]; DataSelect = (Address == 5); "parallel data I/O ControlSelect = (Address == 6); "handshake status/control IRQMaskSelect = (Address == 7); "write interrupt mask bit EQUATIONS " HBCR.sub.-- WE -- write enable for '646. This is a qualified copy of MEMW.sub.--. HBCR.sub.-- WE = DataSelect & MEMW; " HBCR.sub.-- OE -- output enable for '646. This is a qualified copy of MEMR.sub.-- or " the HBCR's transceiver OE.sub.--. This is a hack to use the '646 " like a '652, which was unavailable on short notice. " " Note that the host may read the '646 ONLY when HBCR.sub.-- DWR.sub.-- " is active. Otherwise, the HBCR AD bus may be disturbed. HBCR.sub.-- OE = DataSelect & MEMR # IHBCR.sub.-- PGB & IHBCR.sub.-- PDR; " D1, D0 -- two bit status register for HBCR.sub.-- DWR and HBCR.sub.-- CRD. Note that " ""raw"" values are read: if HBCR.sub.-- DWR.sub.-- is low, D1 will read 0. ENABLE DataBus = ControlSelect & MEMR; D1 = HBCR.sub.-- CRD.sub.-- ; D0 = HBCR.sub.-- DWR.sub.-- ; " HBCR.sub.-- CDY, HBCR.sub.-- DDY -- two bit register for hand-shake outputs. Note that " ""raw"" values are written: if D1 is 0, HBCR.sub.-- CDY.sub.-- will be low. HBCR.sub.-- CDY.sub.-- := ControlSelect & D1 "track # IControlSelect & HBCR.sub.-- CDY.sub.-- ; "hold HBCR.sub.-- DDY.sub.-- := ControlSelect & D0 "track # IControlSelect & HBCR.sub.-- DDY.sub.-- ; "hold " IRQ -- interrupt request. RTS.sub.-- is not currently used, since there is no " mask available to disable it. If HBCR.sub.-- RTS.sub.-- is advisory, " the host can leave HBCR.sub.-- DDY.sub.-- low so that is is always ready " to receive. RTS.sub.-- may be polled externally. IRQ = IIRQMASK & HBCR.sub.-- DDY & HBCR.sub.-- DWR "data ready to read # IIRQMASK & HBCR.sub.-- CDY & HBCR.sub.-- CRD "data ready to write # IIRQMASK & UART.sub.-- RDA; "UART interrupt " IRQMASK -- a one bit mask for IRQ. IRQ is enabled by writing with D0 = 0, " and disabled by writing with D0 = 1. This is redundant, as " the Z86017 PCMCIA adapter IC can also mask the interrupt. " If another output pin is needed, this is the one to take. IRQMASK := IRQMaskSelect & D0 "track #

IIRQMaskSelect & IRQMASK; "hold @if (0) { TEST.sub.-- VECTORS `System Byte Enables` @radix 2; } END hbcr;

CLAIMS:

10. A system comprising:

a host computer which is a palmtop or personal digital assistant, said host computer having a host bus and a PCMCIA slot having a PCMCIA bus therein which is coupled to a system bus of said host computer, said host computer controlled by one or more programs including an operating system program, a PC card interrupt service routine, a card services program, a memory technology device driver program and a client application that needs alphanumeric data encoded into a barcode, magnetic stripe or magnetic ink;

first means for reading said barcode, magnetic stripe or magnetic ink and outputting electrical signals that encode the alphanumeric characters encoded in said barcode, magnetic stripe or magnetic ink;

a PC card inserted in said PCMCIA slot of said host computer and having a PCMCIA bus coupled to said PCMCIA bus of said host computer, said PC card having expansion nonvolatile memory for storing data needed by any program which controls the operation of said host computer, said PC card further comprising second means for receiving said electrical signals from said first means and decoding said electrical signals into alphanumeric characters, storing said alphanumeric characters in a memory on said PC card which is mapped into the I/O space of said PC card, and notifying said host computer that one or more alphanumeric characters are available for reading by generating an interrupt, and

wherein said card services program is structured to control said host computer to assign an interrupt number and I/O space including a base address to said PC card and store said data, and wherein said memory technology driver program conrols said host computer to register itself with said card services program as a client and interrogate said card services program and determine said interrupt number and said base address in the I/O space assigned to said PC card, and wherein said client application includes means for controlling said host computer to read said alphanumeric characters from said memory on said PC card by registering said client application with said card services program, and by interrogating by any interprocess data transfer mechanism, said card services program on a client by client basis, each interrogation including passing a predetermined subfunction argument number to said card services program which is the client number reserved for said memory technology driver program when it registered with said card services program and by interpreting the data returned in response to each said interrogation to determine if the returned data has at a first offset a signature unique to said memory technology driver program, and, if not, controlling said computer to repeat the interrogation for the next client, and if the returned data in response to the interrogation contains the unique signature, then the returned data indicates the client registered to said subfunction argument number is said memory technology driver program and the data at a second offset in said returned data is the interrupt number assigned to said PC card and the base address of the I/O space of said PC card, and controlling said host computer to read said interrupt number and base address from said second offset in said returned data, and said client application program passing the interrupt number and the start address of said PC card interrupt service routine to said operating system program for storage in an interrupt vector table maintained by said operating system for use as the interrupt number of said PC card and to define in said table the start address of the interrupt service routine used to retrieve said alphanumeric characters decoded by said second means stored in memory in the I/O space of said PC card; and

wherein said second means in said PC card includes a PCMCIA adapter circuit which activates an interrupt signal line on said PCMCIA bus when a decoded alphanumeric character is ready to be retrieved from the I/O space memory of said PC card;

and wherein said host computer includes an interrupt controller circuit coupled to said interrupt signal line of said PCMCIA bus which functions to detect said activation of said interrupt signal line and place the interrupt number assigned to said PC card on said host bus;

and wherein said operating system is structured to control said host computer to use said interrupt number as an index into said interrupt vector table to retrieve the start address of said PC card interrupt service routine and invoke it into execution;

and wherein said PC card interrupt service routine is structured to control said host computer to execute an I/O cycle and use said base address of the I/O space of said PC card to retrieve at least one alphanumeric character stored in memory in the I/O space of said PC card.

16. A system comprising:

a host computer in the form of a personal digital assistant or palmtop computer, said host computer having a PCMCIA slot having a PCMCIA bus connector therein with a PCMCIA card engaged in said PCMCIA slot and electrically coupled to said host computer through said PCMCIA bus, said PCMCIA card having a data storage memory for storing sample data, said host computer including decoding means for receiving sample data of a barcode pattern via said PCMCIA bus connector and decoding said sample data into one or more alphanumeric characters and storing said alphanumeric characters in a memory in said host computer, said host computer including a PCMCIA bus controller circuit coupled to receive said barcode image sample data via said PCMCIA bus connector and supply it to a barcode decoding process carried out by said decoding means and coupled to detect an interrupt generated by said PCMCIA card and output an interrupt number assigned to said PCMCIA card, said decoding means including sample data retrieval means for controlling said host computer to assign an interrupt number to said PC card and map said data storage memory into the I/O space of said PCMCIA card and assign a base address in the I/O space of said host computer to said I/O space of said PCMCIA card, and to indirectly interrogate a memory technology driver program controlling said host computer via a registered client by registered client interrogation process of a card services program controlling said host computer so as to determine the interrupt number and base address of the I/O space of said PCMCIA card and store them in an interrupt vector table maintained by an operating system also controlling said host computer and to store the start address of a PCMCIA card interrupt service routine in said interrupt vector table, and for controlling said host computer to use the interrupt number of said PCMCIA card interrupt to find the base address of the I/O space of said PCMCIA card and invoke said PCMCIA card proper interrupt service routine and use said base address to retrieve sample data from said PCMCIA card via I/O cycles carried out across said PCMCIA bus connector;

and wherein said PCMCIA card comprises:

a PCMCIA bus connector that mates with said PCMCIA bus connector in said PCMCIA slot of said host;

a nonvolatile flash EEPROM expansion memory mapped into the memory space of said host computer;

an undecoded barcode scan engine either integrated on or mounted on said PCMCIA card and outputting an electrical signal at an output that encodes the light and

dark patterns of any barcode scanned by said scan.

17. A circuit comprising:

a host computer in the form of a palmtop or personal digital assistant and having a system bus and a PCMCIA bus in a PCMCIA slot and coupled to said system bus via a PCMCIA bus controller circuit, said host computer controlled by a barcode decoding program to receive undecoded barcode sample data and decode alphanumeric data encoded therein, said host computer further controlled by an operating system, a PCMCIA card interrupt service routine dedicated to retrieving said undecoded barcode sample data from said PCMCIA card, and further programmed by said PCMCIA card management software which controls said host computer to detect the presence of a PCMCIA card in said PCMCIA slot and map one or more memory locations on said PCMCIA card which store said undecoded barcode sample data into the I/O space of said host computer using a base address and to assign an interrupt number for said PCMCIA card and to store said interrupt number assigned to said PCMCIA card and a start address for said PCMCIA card interrupt service routine in an interrupt vector table used by said operating system, and to store said base address in a location known to said PCMCIA card interrupt service routine, said PCMCIA bus controller circuit for detecting an interrupt signal generated by said PCMCIA card and notifying said operating system of the interrupt number of said PCMCIA card, said operating system for controlling said host computer to use said interrupt number to execute said PCMCIA card interrupt service routine, said PCMCIA interrupt service routine using I/O cycles and said base address to retrieve said sample data and pass it to said barcode decoding program for decoding into alphanumeric characters, said host computer further programmed by one or more programs which control said host computer to read data from and write data for any purpose to any auxiliary flash EEPROM on a PCMCIA card engaged in said PCMCIA slot, said auxiliary flash EEPROM being mapped by said PCMCIA card management software into the memory space of said host computer, said one or more programs structured to access said auxiliary flash EEPROM using memory cycles without rebooting said host after use of I/O cycles to retrieve said sample data;

a PCMCIA card engaged in said PCMCIA slot of said host computer, comprising:

a PCMCIA bus coupled to said PCMCIA bus of said host computer;

a barcode scan engine of the laser or wand type either mounted on said PCMCIA card of a stand alone variety not mounted on said PCMCIA card but coupled to an input port on said PCMCIA card and outputting undecoded signals encoding alphanumeric characters;

a PCMCIA adapter chip coupled to said PCMCIA bus and coupled to said barcode scan engine and functioning to receive said undecoded signals from said barcode scan engine, generate samples thereof and transfer said samples to a host computer through said PCMCIA bus, and having a latch, shift register or other data storage entity that stores said sample data and is mapped by said card management software of said host into the I/O space of said host at a known location at or in some known relationship to said base address; and

an auxiliary flash EEPROM memory which is mapped by software controlling said host computer into the memory space of said host computer.

18. A circuit comprising:

a host computer having a system bus and a PCMCIA slot with a PCMCIA bus therein coupled via a PCMCIA bus controller circuit to said system bus and controlled by, said host computer programmed with and controlled by a plurality of software programs including an operating system, a barcode client application, a PCMCIA card interrupt service routine dedicated to retrieving from said PCMCIA card decoded

alphamumeric characters from any PCMCIA card engaged in said PCMCIA slot, and further controlled by PCMCIA card management software which controls said host computer to detect the presence of a PCMCIA card in said PCMCIA slot, assign an interrupt number thereto and map a data register on any PCMCIA card engaged in said PCMCIA slot into the I/O space of said host computer at a known location relative to a base address in said I/O space of said host computer assigned to a memory location or storage register on said PCMCIA card, said data register functioning to store the alphnumeric character to be retrieved, and wherein said barcode client application controls said computer to interrogate said PCMCIA card management software to determine said interrupt number and said base address and store said interrupt number and a start address for said PCMCIA card interrupt service routine in an interrupt vector table used by said operating system, said barcode client application also controlling said computer to store said base address in a location known to said PCMCIA card interrupt service routine, said PCMCIA card management software further controlling said host computer to map one or more other registers and any auxiliary nonvolatile flash EEPROM memory on said PCMCIA card into the memory space of said host computer, said PCMCIA bus controller circuit for detecting an interrupt signal generated by said PCMCIA card and notifying said operating system of the interrupt number of said PCMCIA card, said operating system for controlling said host computer to use said interrupt number to find and invoke into execution said PCMCIA card interrupt service routine dedicated to retrieving alphanumeric characters from said PCMCIA card, said PCMCIA interrupt service routine and said PCMCIA card managment software for controlling said host computer to write data to predetermined control registers on said PCMCIA card to force said PCMCIA card into an I/O mode and controlling said host computer to use I/O cycles and said base address to retrieve said alphanumeric character from said data register and pass said alphanumeric character to said barcode client application for use, said host computer further programmed by one or more programs which control said host computer to read data from and write data for any purpose to any auxiliary nonvolatile flash EEPROM memory on a PCMCIA card engaged in said PCMCIA slot, said one or more programs structured to control said host computer along with said PCMCIA card management software to force said PCMCIA card into a memory mode without rebooting said host computer and to access said auxiliary flash EEPROM nonvolatile memory using memory cycles;

a PCMCIA card comprising:

a PCMCIA bus coupled to said PCMCIA bus of said host computer;

a barcode scan engine of the laser or wand type mounted on said PCMCIA card and outputting undecoded signals encoding alphanumeric characters;

a barcode decoder circuit having an input coupled to receive undecoded signals encoding alphanumeric characters embedded in a barcode and functioning to decode said undecoded signals into one or more alphanumeric characters and presenting said alphanumeric character(s) at an output;

data register coupled to said PCMCIA bus and coupled to said barcode decoder circuit for receiving and temporarily storing said one or more alphanumeric character(s) decoded by said barcode decoder circuit until said PCMCIA interrupt service routine and said PCMCIA card management software control said host computer to force said PCMCIA card into I/O mode perform one or more I/O cycles to read the alphanumeric character(s) to be retrieved;

an auxiliary nonvolatile flash EEPROM memory segmented into a high byte and a low byte for use at least as expansion or auxiliary memory for said host computer and mapped into memory space of said host computer by said PCMCIA card management software;

byte steering means coupling both said high and low byte segments of said auxiliary

nonvolatile flash EEPROM memory to high and low bytes of a data portion of said PCMCIA bus for steering the high byte data stored in said high byte segment of said auxiliary nonvolatile flash EEPROM memory selectably onto either said high byte or said low byte of said data portion of said PCMCIA bus, and for coupling said low byte segment of said auxiliary nonvolatile flash EEPROM memory to said low byte of said data portion of said PCMCIA bus;

a card information means coupled to said PCMCIA bus and mapped into the attribute space of said PCMCIA card and into the memory space of said host computer, for storing card information structure data defining how said PCMCIA card is organized;

a configuration option register means coupled to said PCMCIA bus and mapped into the attribute space of said PCMCIA card and into the memory space of said host computer, for storing a mode bit written into said configuration option register means by said host computer, said mode bit indicating whether said host computer desires said PCMCIA card to operate in I/O mode or memory mode such that other circuits can determine the meanings of predetermined signals on said PCMCIA bus that have different meanings in said I/O mode and said memory mode;

card status register means coupled to said PCMCIA bus and mapped into said attribute space of said PCMCIA card for storing an interrupt enable mask bit written into said card status register means by said host computer such that said host computer can disable interrupts from said PCMCIA card;

pin replace register means coupled to said PCMCIA bus and mapped into said attribute space of said PCMCIA card and into the memory space of said host computer, for storing the memory mode states of the signals on certain shared pins of said PCMCIA bus when said PCMCIA card was operating in memory mode for use by other circuits on said PCMCIA card that need the states of these signals for memory mode transactions such as accessing said EEPROM memory when said PCMCIA card is operating in I/O mode such that said host computer does not have to be rebooted after operating said PCMCIA card in I/O mode to retrieve decoded alphanumeric characters in order to operate in memory mode to access data stored in said nonvolatile memory on said PCMCIA card; control register means coupled to said PCMCIA bus for storing one or more pacing bits needed to signal status of said host for a handshaking protocol between said host computer and said PCMCIA bus to carry out I/O send and receive transactions with said data register means or other circuits on said PCMCIA card, said pacing signal bits stored therein by said host computer under control of said barcode client application or said PCMCIA card interrupt service routine;

status register means coupled to said PCMCIA bus for storing PC card status bits the states of which are controlled by the assertion of one or more signals output by said decoder circuit, said PC card status bits being read by said host computer under control of said barcode client application or said PCMCIA card interrupt service routine thereby causing said host computer to write said pacing bits into said control register means as part of said handshaking protocol of an I/O transaction to retrieve a decoded alphanumeric character. and wherein said control and logic interrupt means activates said PCMCIA card interrupt signal to said host computer based upon a predetermined Boolean logic combination of said bits stored in said status register means and said control register means;

command register means coupled to said PCMCIA bus, for storing commands generated by said host computer under control of said barcode client application to control the manner in which said decoder circuit means carries out said decoding process so as to provide options to add specified prefixes or suffixes;

an input port to receive data from any undecoded barcode scanning device external to said PCMCIA card; and

a multiplexer having a first input coupled to received undecoded signals from said barcode scan engine and having a second input coupled to receive undecoded signals from said input port and having an output coupled to said input of said barcode decoder circuit, for coupling either said signals from said barcode scan engine or signals from said input port to said barcode decoder circuit under the control of a bit set in said control register.

19. A process executed on a host computer and a PCMCIA card, said host computer having a PCMCIA card slot engaged with a PCMCIA card having at least a barcode decoder circuit and nonvolatile auxiliary memory thereon, comprising:

booting said host computer to put an operating system into execution, and executing one or more PCMCIA card management programs and a barcode driver program on said host computer to control said host computer to find and manage said PCMCIA card and assign an interrupt number thereto and map various data storage registers and said nonvolatile auxiliary memory on said PCMCIA card into a memory address space of said host computer and to map one or more data registers on said PCMCIA card that store alphanumeric characters output by said barcode decoder circuit into an I/O address space of said host computer starting at an assigned base address in said I/O space and to store said interrupt number and the start address of a PCMCIA interrupt service routine which controls said host computer to retrieve said alphanumeric characters from said PCMCIA card in an interrupt vector table used by said operating system to launch interrupt service routines, and to store said base address of said one or more data registers in said I/O space in a memory on said host computer in such a way that said base address can be retrieved by said PCMCIA interrupt service routine, said PCMCIA card either having data written to one or more of said various data storage registers by said PCMCIA card management programs or defaulting to a data state at boot time to force said PCMCIA card into memory mode such that said auxiliary nonvolatile memory can be accessed via memory cycles;

receiving a command from a user to launch a barcode application program that needs alphanumeric data entered by keyboard or through scanning a barcode, and responding thereto by loading and executing an application program to control said host computer to use keyboard data entered by a keyboard of said host computer or other alphanumeric data retrieved from said PCMCIA card such as alphanumeric data decoded by said PCMCIA card from the signals output by a barcode scan engine;

receiving a command from said user by entering a command to said barcode application program or to said operating system to scan a barcode and passing a message to said barcode driver program indicating scanning of a barcode is desired;

controlling said host computer using said barcode driver program so as to send a start signal to said PCMCIA card indicating scanning of a barcode is desired;

receiving said start signal on said PCMCIA card and sending a start signal to a barcode scan engine or applying power thereto so as to activate scanning;

detecting signals arriving from said barcode scan engine at said barcode decoder circuit and determining the type of barcode being scanned and the direction of the scan and decoding one or more alphanumeric characters from the signals arriving from said barcode scan engine and storing said alphanumeric characters in said one or more data resisters on said PCMCIA card mapped into the I/O space of said host computer;

generating an interrupt signal on said PCMCIA card and sending said interrupt signal to said host computer after at least one alphanumeric character has been successfully decoded; and

executing said PCMCIA interrupt service routine on said host computer and one or more of said PCMCIA card management programs to control said host computer to send data to predetermined data storage registers on said PCMCIA card to force said card into I/O mode and to execute one or more I/O cycles to retrieve said decoded alphanumeric character or characters from said PC card and do whatever is desired with said character such as pass the character to said application program or place said character in a keyboard buffer memory of said host computer or write the character into a nonvolatile memory on said PC card while using one or more of said various data storage registers on said PCMCIA card to store the signal state in said memory mode of various signals on said PCMCIA bus that are on pins that carry first predetermined signals in memory mode and second predetermined signals in I/O mode such that after said I/O cycles are completed, memory cycles can be resumed to access said auxiliary nonvolatile memory without rebooting said host computer; and

using memory cycles to access said auxiliary nonvolatile memory during intervals when said PCMCIA bus is not tied up with performing I/O cycles.

21. A system comprising:

a PCMCIA card having a PCMCIA bus connector and either an undecoded barcode scan engine integrated or mounted thereon which generates an undecoded barcode scan signal that encodes the light and dark patterns of a barcode or having an input port for receiving an undecoded barcode scan signal that encodes the light and dark patterns of a barcode from a barcode scanning system which is external to said PC card and coupled to said input port by an electrical cable, and further comprising a sampling and PCMCIA interface adapter means coupled to receive said undecoded barcode scan signal and coupled to said PCMCIA bus connector, for generating compressed sample data of said undecoded barcode scan signal by determining a count value in a running count at the time of every transistion of said undecoded barcode scan signal above or below a reference voltage level and storing said counts in a latch or FIFO memory and generating an interrupt at predetermined times;

a host computer in the form of a palmtop or personal digital assistant having a PCMCIA slot having said PCMCIA card engaged therewith with the PCMCIA bus connector of said PCMCIA card in electrical contact with a PCMCIA bus connector in said PCMCIA slot, said host computer having a PCMCIA bus controller circuit coupled to said PCMCIA bus for receiving said interrupt signal from said PCMCIA card and putting an interrupt number assigned to said PCMCIA card on a host bus of said host computer and for reading data from a data bus which is part of said PCMCIA bus and making it available directly or indirectly to a barcode decoding process, said host computer programmed with software to control said host computer to detect the presence of said PCMCIA card and assigning an interrupt number to said PCMCIA card and mapping said latch or FIFO memory of said PCMCIA card into either the I/O space or memory space of said host computer, and for controlling said computer to execute a predetermined interrupt service routine when said interrupt of said PCMCIA card is detected, said interrupt service routine controlling said computer to read said latch or FIFO memory of said PCMCIA card using either memory or I/O cycles and store the count stored in said latch or FIFO memory in a memory of said host computer, said software of said host computer also structured to control said host computer to decode alphanumeric characters encoded in said undecoded barcode scan signal, as determined by said counts retrieved from said latch or FIFO memory on said PCMCIA card and using said run lengths to decode said alphanumeric signals.

Previous Doc Next Doc Go to Doc#

Previous Doc Next Doc Go to Doc#
First Hit

Generate Collection

L10: Entry 16 of 17 File: DWPI Nov 16, 2002

DERWENT-ACC-NO: 2001-125669
DERWENT-WEEK: 200302

EP1 065 626 A1

TITLE: Memory management procedure for a mobile telephone SIM or chip card of type 2+ where all data in volatile memory is backed up to an internal or external memory prior to the card being put in standby mode

Basic Abstract Text (1):
NOVELTY - Procedure has the following steps: a command is sent to the chip card (2) to enter into a standby state, the chip card sends a reply to the request and the card is put in standby state depending on the response. The card is only put on standby once data in its RAM has been backed up and the chip card decides whether to back up the data to an internal or external memory dependent on its activity.

PF Application Date (1):
20000615

PF Application Date (3):
20000615

PF Application Date (4):
19990630

PF Application Date (5):
20000615

PF Application Date (6):
20000615

PF Application Date (7):
20000615

Equivalent Abstract Text (1):
NOVELTY - Procedure has the following steps: a command is sent to the chip card (2) to enter into a standby state, the chip card sends a reply to the request and the card is put in standby state depending on the response. The card is only put on standby once data in its RAM has been backed up and the chip card decides whether to back up the data to an internal or external memory dependent on its activity.

Standard Title Terms (1):
MEMORY MANAGEMENT PROCEDURE MOBILE TELEPHONE CHIP CARD TYPE DATA VOLATILE MEMORY BACK UP INTERNAL EXTERNAL MEMORY PRIOR CARD STANDBY MODE

Previous Doc Next Doc Go to Doc#

Previous Doc Next Doc Go to Doc#
First Hit

Generate Collection

L5: Entry 1 of 19 File: PGPB Mar 20, 2003

DOCUMENT-IDENTIFIER: US 20030055870 A1
TITLE: REMOTE BOOKMARKING FOR WIRELESS CLIENT DEVICES

Application Filing Date:
19990122

Detail Description Paragraph:
[0037] Wireless client devices, also referred to as two-way interactive communication or mobile devices, include but are not limited to personal digital assistant (PD A) like devices, cellular phones, or wireless capable remote controllers. Such devices typically have significantly less memory and processing capability than is found in desktop and laptop computers. These wireless client devices, which are not a combination of a computer and a wireless communications module, have a small display screen and a limited keypad as opposed to the keyboards associated with desktop computers.

Detail Description Paragraph:
[0040] A proxy server device 116 is coupled between landnet 120 and airnet 112. The proxy server device 116 is also known as a network gateway server. The proxy server device 116 can be implemented as a workstation computer or a personal computer. Often, the communication protocol used in airnet 112 is different from that used in landnet 120. As a result, one of the functions that proxy server device 116 performs is to map or translate from one communication protocol to another, thereby wireless device 100 coupled to airnet 112 can communicate with any of the information server devices (e.g. information server 124) coupled to landnet 120 via proxy server device 116. The proxy server device 116 also store or provide access to accounting services, configuration management services, and dedicated storage for applications and files for user accounts. These applications and services may be resident on proxy server device 116 or on a separate server device accessible via landnet 120.

Detail Description Paragraph:
[0043] It should be noted that HDTP is a session-level protocol that resembles HTTP but without incurring the overhead thereof and is highly optimized for use in thin devices, such as mobile devices that have significantly less computing power and memory than a desktop personal computer. Further, it is understood to those skilled in the art that UDP does not require a connection to be established between a client and a server before information can be exchanged, which eliminates the need of exchanging a large number of packets during a session creation between a client and a server. Exchanging a very small number of packets during a transaction is a desired feature for a mobile device with very limited computing power and memory to effectively interact with a landline device.

Previous Doc Next Doc Go to Doc#

Previous Doc Next Doc Go to Doc#
First Hit Fwd Refs

Generate Collection

L5: Entry 4 of 19 File: USPT May 6, 2003

DOCUMENT-IDENTIFIER: US 6560640 B2
TITLE: Remote bookmarking for wireless client devices

Application Filing Date (1):
19990122

Detailed Description Text (5):
Wireless client devices, also referred to as two-way interactive communication or mobile devices, include but are not limited to personal digital assistant (PDA) like devices, cellular phones, or wireless capable remote controllers. Such devices typically have significantly less memory and processing capability than is found in desktop and laptop computers. These wireless client devices, which are not a combination of a computer and a wireless communications module, have a small display screen and a limited keypad as opposed to the keyboards associated with desktop computers.

Detailed Description Text (8):
A proxy server device 116 is coupled between landnet 120 and airnet 112. The proxy server device 116 is also known as a network gateway server. The proxy server device 116 can be implemented as a workstation computer or a personal computer. Often, the communication protocol used in airnet 112 is different from that used in landnet 120. As a result, one of the functions that proxy server device 116 performs is to map or translate from one communication protocol to another, thereby wireless device 100 coupled to airnet 112 can communicate with any of the information server devices (e.g. information server 124) coupled to landnet 120 via proxy server device 116. The proxy server device 116 also store or provide access to accounting services, configuration management services, and dedicated storage for applications and files for user accounts. These applications and services may be resident on proxy server device 116 or on a separate server device accessible via landnet 120.

Detailed Description Text (11):
It should be noted that HDTP is a session-level protocol that resembles HTTP but without incurring the overhead thereof and is highly optimized for use in thin devices, such as mobile devices that have significantly less computing power and memory than a desktop personal computer. Further, it is understood to those skilled in the art that UDP does not require a connection to be established between a client and a server before information can be exchanged, which eliminates the need of exchanging a large number of packets during a session creation between a client and a server. Exchanging a very small number of packets during a transaction is a desired feature for a mobile device with very limited computing power and memory to effectively interact with a landline device.

Previous Doc Next Doc Go to Doc#

Previous Doc Next Doc Go to Doc#
First Hit Fwd Refs

Generate Collection

L10: Entry 12 of 17 File: USPT Nov 26, 1996

DOCUMENT-IDENTIFIER: US 5579489 A
TITLE: Hand-held portable computer having capability for external expansion of an internal bus

Abstract Text (1):
A personal digital assistant module with a local CPU, memory, and I/O interface has a host interface comprising a bus connected to the local CPU and a connector at a surface of the personal digital assistant for interfacing to a bus connector of a host general-purpose computer, providing direct bus communication between the personal digital assistant and the host general-purpose computer. In an embodiment, the personal digital assistant also stores a security code. The personal digital assistant according to the invention forms a host/satellite combination with a host computer having a docking bay, wherein upon docking a docking protocol controls access by the host to memory of the personal digital assistant based on one or more passwords provided by a user to the host. In another embodiment the personal digital assistant also has an expansion port connected to the local CPU, and expansion peripheral devices may be connected and operated through the expansion port.

Application Filing Date (1):
19940726

Drawing Description Text (23):
FIG. 21 is a largely schematic representation of a computer with an external disk storage device connected through a PIO port and signal cable according to the present invention.

Previous Doc Next Doc Go to Doc#

Previous Doc Next Doc Go to Doc#
First Hit

Generate Collection

L10: Entry 14 of 17 File: DWPI Mar 16, 2004

DERWENT-ACC-NO: 2002-364348
DERWENT-WEEK: 200420

EP1182560 A2

TITLE: Processor and memory system especially for use with embedded systems such as mobile phones, PDAs, etc. where an additional flash memory containing program code is connected and a memory management system optimizes memory access

Basic Abstract Text (1):
NOVELTY - System comprises a processor component (2) with a processor (6) and one or more integrated RAM memories (9). A second memory (3) is coupled to the processor via an interface (5). A memory management device (7) checks whether a data address corresponds to the fast integral RAM or the external memory. If the data address corresponds to the fast RAM then the data is read from it, while if the address is not a RAM address an interrupt command is issued and the data found and loaded from the external memory.

Basic Abstract Text (2):
USE - Processor memory system for use with mobile phones, PDAs, etc. where typically a processor and integral or fast memory are complemented by an external memory in which a program is stored. Such configurations are found in embedded systems.

Basic Abstract Text (6):
flash external memory 3

PF Application Date (3):
20000817

Equivalent Abstract Text (1):
NOVELTY - System comprises a processor component (2) with a processor (6) and one or more integrated RAM memories (9). A second memory (3) is coupled to the processor via an interface (5). A memory management device (7) checks whether a data address corresponds to the fast integral RAM or the external memory. If the data address corresponds to the fast RAM then the data is read from it, while if the address is not a RAM address an interrupt command is issued and the data found and loaded from the external memory.

Equivalent Abstract Text (2):
USE - Processor memory system for use with mobile phones, PDAs, etc. where typically a processor and integral or fast memory are complemented by an external memory in which a program is stored. Such configurations are found in embedded systems.

Equivalent Abstract Text (6):
flash external memory 3

Standard Title Terms (1):
PROCESSOR MEMORY SYSTEM EMBED SYSTEM MOBILE TELEPHONE ADD FLASH MEMORY CONTAIN PROGRAM CODE CONNECT MEMORY MANAGEMENT SYSTEM OPTIMUM MEMORY ACCESS

Previous Doc Next Doc Go to Doc#
First Hit Fwd Refs

Generate Collection

L10: Entry 4 of 17 File: USPT Apr 22, 2003

DOCUMENT-IDENTIFIER: US 6553468 B1
TITLE: Storage method for personal digital devices

Abstract Text (1):
In a device having data storage, storage space is adaptively controlled by monitoring what pre-stored instructional information is used and how the pre-stored instructional information is used. Some or all of the storage space containing the pre-stored instructional information may be overwritten with user data based on the adaptive control. However, some instructional information may be designated as not overwritable. If access to an external storage device is detected, such as via a communications network to a remote web site, then the pre-stored instructional information can be partially or completely overwritten and instructional information accessed when needed from the external storage device. If additional storage space is still required for user data after all the overwritable pre-stored instructional information has been overwritten, then user data may be transferred to the external storage device for remote storage. The overwriting may be automatic or interactive.

Application Filing Date (1):
20000626

Brief Summary Text (28):
According to another aspect of the invention, when all instructional information has already been overwritten with user data, and additional storage space is still required, a warning is issued, and then a check is made to determine if the device has is "web-aware," that is, has access to storage at a remote web site, or other external storage device.

Brief Summary Text (30):
According to another aspect of the invention, where the device is not "web-aware," when the available storage space on-board has been used up, user data can be transferred to an external data storage device if available. For example, the device may have a port for transferring data to a portable disk drive, such as a zip drive, for external storage.

Detailed Description Text (20):
Where the device does not have web access, user data could be transferred to an external data storage device, if available. For example, the device may have a serial or parallel port for transferring data to a portable disk drive, such as a zip drive, for external storage.

Detailed Description Text (24):
Further, in the case where an external data storage device is available to the user, such as an external zip drive, or the like, rather than simply overwriting the pre-stored instructional information in any of the above-mentioned steps, an option would be that the instructional information could be transferred to the external storage for later access, i.e., backed-up, before the on-board storage space is overwritten with user data.

Detailed Description Text (29):
FIG. 4 illustrates a simplified block diagram of an arrangement according to an exemplary embodiment of the invention. A personal digital device 400 has on-board storage 401 and an on-board control 402 for managing the storage and access of user data and instructional information for the device 400. The device 400 may be connectable via an input/output (I/O) port or link 403 to an external storage device 404 which can store either or both of instructional information and user data, as indicated by block 405. In addition, or alternatively, the device 400 may have access to a network link 406 to a remote support web site represented by block 407, e.g., over the Internet, where storage of instructional information and/or user data is possible as represented by block 408.

Detailed Description Text (32):
This intelligent overwriting is based on what operational features on a specific device have already been mastered by a user, what features are not typically used or will probably not be used, and what access the device has to a communications network (or other external storage device) where the information might easily be accessed, and/or user data stored, rather than from the storage means on the device. The invention is useful with new and/or recyclable devices.

CLAIMS:

7. The method according to claim 6, further comprising: detecting when the total storage area is full of user information; and when it is detected that the total storage area is full of user information: issuing a warning; checking for the availability of external storage; and if external storage is available, then transferring user data to the external storage.

11. The method according to claim 1, further comprising: checking for the availability of external storage; and if external storage is accessible, then before the selectively overwriting, transferring the instructional information to the external storage and storing the instructional information in the external storage for later access.

15. An arrangement comprising: a personal digital device having on-board storage, the on-board storage including user data storage space and pre-stored instructional information storage space; control means in the personal digital device, for controlling user data storage; wherein the control means controls user storage so that: the instructional information is divided into basic and advanced information; a frequency of access to the instructional information is detected and a user proficiency level determined based on the premise that the higher the frequency of access to the instructional information, the more novice the user; if the user proficiency level is determined to be novice, then the advanced instructional information is deleted; and if the user proficiency is determined to be advanced, then the basic instructional information is deleted.

17. The arrangement according to claim 16, wherein the control means further controls the user storage so that: if the on-board storage is full, and additional user data storage space is required, then: a warning is issued; access to an external storage is checked; and if access to an external storage is available, then user data is transferred to the external storage.

18. The arrangement according to claim 17, wherein the external storage comprises an external direct access storage device.

19. The arrangement according to claim 18, wherein the external storage comprises an external high-capacity removable disk device.

20. The arrangement according to claim 17, wherein the external storage comprises a remote web site.